APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
1[a]. A speech system using an analysis by synthesis approach on a speech signal, the speech system comprising:	According to publicly available documentation, [Defendants' Accused Products][1] implement a speech system using an analysis by synthesis approach on a speech signal. The 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Mandatory speech CODEC speech processing functions; AMR speech CODEC; General description ("TS 26.071") describes the mandatory speech codec speech processing functions for the Adaptive Multi-Rate ("AMR") speech codec. "The multi-rate speech coder is a single integrated speech codec with eight source rates from 4.75 kbit/s to 12.2 kbit/s, and a low rate background noise encoding mode." TS 26.071 at § 4. The 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Mandatory Speech Codec speech processing functions; Adaptive Multi-Rate (AMR) speech codec; Transcoding functions ("TS 26.090") describes the mandatory transcoding functions for the Adaptive Multi-Rate ("AMR") speech codec. "Clauses 4.3 and 4.4 present a simplified description of the principles of the AMR codec encoding and decoding process respectively. In clause 4.5, the sequence and subjective importance of encoded parameters are	To the extent this limitation is satisfied by the functionality accused in Plaintiff's 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard includes the following passages, which are virtually identical to the passages identified by WIAV: 4.3 Principles of the GSM enhanced full rate speech encoder The codec is based on the code-excited linear predictive (CELP) coding model. A 10th order linear prediction (LP), or short-term, synthesis filter is used which is given by: **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 4.3 (Mar. 1997)).**

[1] Plaintiff's infringement contentions are virtually identical for each of the Defendants' respective accused products. To illustrate the similarity between the functionality accused by Plaintiff and that present in the prior art, Defendants have included Plaintiff's contentions herein, but have made replaced the list of accused products in each of Plaintiff's respective infringement contentions with the term "Defendants' Accused Products" for simplicity.

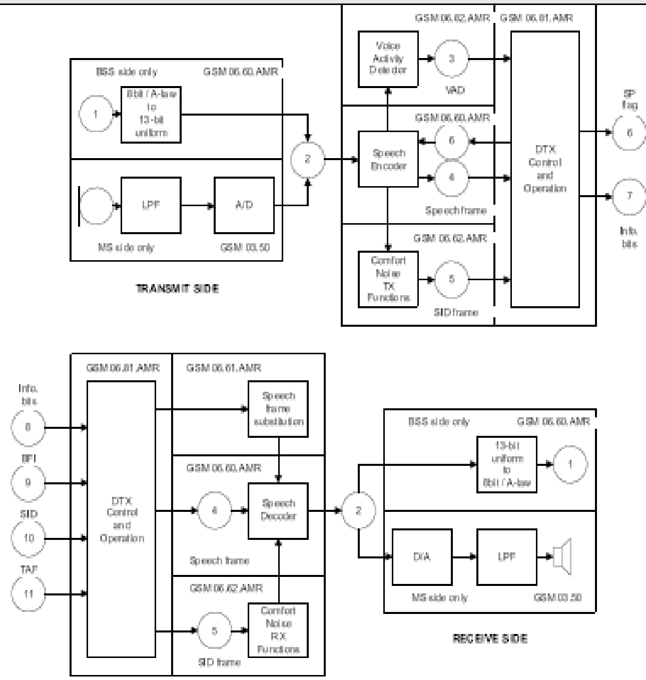
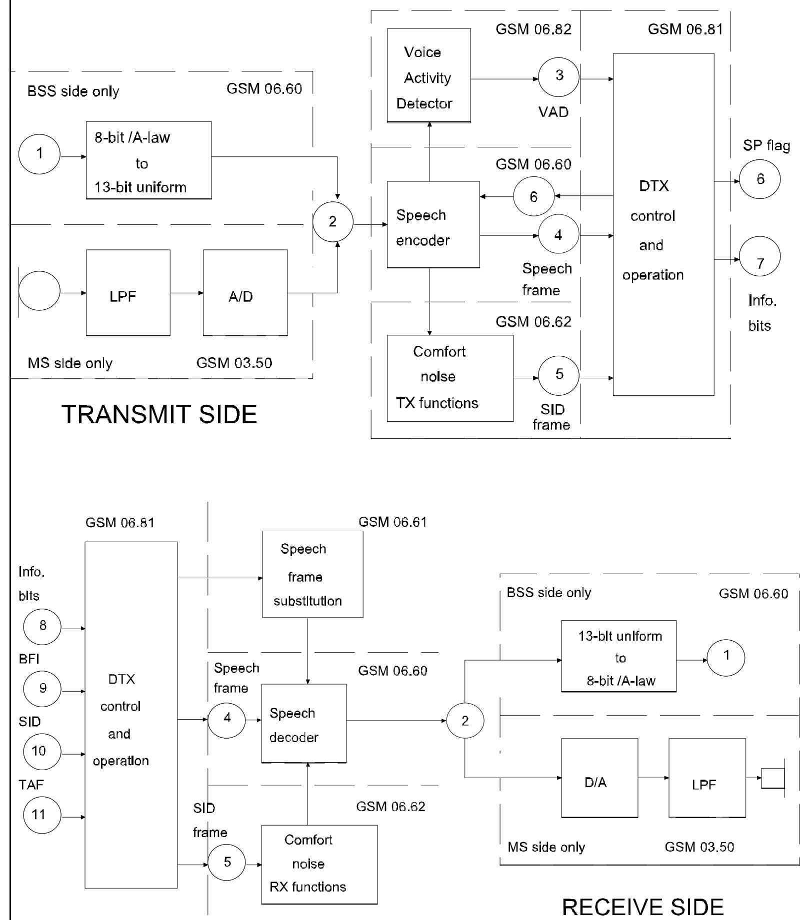
APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	given. Clause 5 presents the functional description of the AMR codec encoding, whereas clause 6 describes the decoding procedures. In clause 7, the detailed bit allocation of the AMR codec is tabulated." TS 26.090 at § 4; § 6 ("The function of the decoder consists of decoding the transmitted parameters (LP parameters, adaptive codebook vector, adaptive codebook gain, fixed codebook vector, fixed codebook gain) and performing synthesis to obtain the reconstructed speech. The reconstructed speech is then post-filtered and upscaled. The signal flow at the decoder is shown in figure 4."). The 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; ANSI-C code for the Adaptive Multi Rate (AMR) speech codec ("TS 26.073") "contains an electronic copy of the ANSI-C code for the Adaptive Multi-Rate codec. The ANSI-C code is necessary for a bit exact implementation of the Adaptive Multi Rate speech transcoder (TS 26.090 [2]), Voice Activity Detection (TS 26.094 [6]), comfort noise (TS 26.092 [4]), source controlled rate operation (TS 26.093 [5]) and example solutions for substituting and muting of lost frames (TS 26.091 [3])." TS 26.071 at § 1. "In the case of discrepancy between the requirements described in the present document and the fixed point computational description (ANSI-C code) of these requirements contained in [TS 26.073], the description in [TS 26.073] will prevail." TS 26.090 at § 1.	The function of the decoder consists of decoding the transmitted parameters (LP parameters, adaptive codebook vector, adaptive codebook gain, fixed codebook vector, fixed codebook gain) and performing synthesis to obtain the reconstructed speech. The reconstructed speech is then post-filtered and upscaled. The signal flow at the decoder is shown in figure 4. **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference Section 6 GSM EFR 06.60 (Mar. 1997)).**

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	[Defendants' Accused Products] include an Adaptive Multi-Rate ("AMR") speech system that uses an analysis by synthesis approach for coding and decoding speech signal. *See e.g.,* TS 26.090 at § 4.3 ("The CELP speech synthesis model is shown in figure 2. In this model, the excitation signal at the input of the short-term LP synthesis filter is constructed by adding two excitation vectors from adaptive and fixed (innovative) codebooks. The speech is synthesized by feeding the two properly chosen vectors from these codebooks through the short-term synthesis filter. The optimum excitation sequence in a codebook is chosen using an analysis-by-synthesis search procedure in which the error between the original and synthesized speech is minimized according to a perceptually weighted distortion measure."), Figure 2; *see also e.g.,* TS 26.071 at § 4; Figure 1 (shown below).	The CELP speech synthesis model is shown in figure 2. In this model, the excitation signal at the input of the short-term LP synthesis filter is constructed by adding two excitation vectors from adaptive and fixed (innovative) codebooks. The speech is synthesized by feeding the two properly chosen vectors from these codebooks through the short-term synthesis filter. The optimum excitation sequence in a codebook is chosen using an analysis-by-synthesis search procedure in which the error between the original and synthesized speech is minimized according to a perceptually weighted distortion measure. **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 4.3 (Mar. 1997)).**

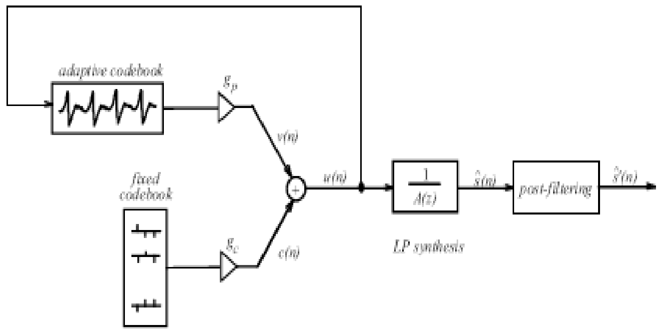
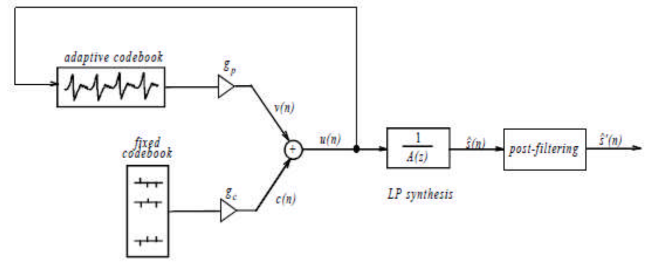
APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	 Figure 1: Overview of audio processing functions	Plaintiff's audio processing Figure 1 does not directly relate to the claims of the '992 patent. However, the same figure can also be found throughout the prior art GSM standards, e.g., GSM 06.51, § 4, Fig. 1, v.4.0.1 (Dec. 1997). See below: 

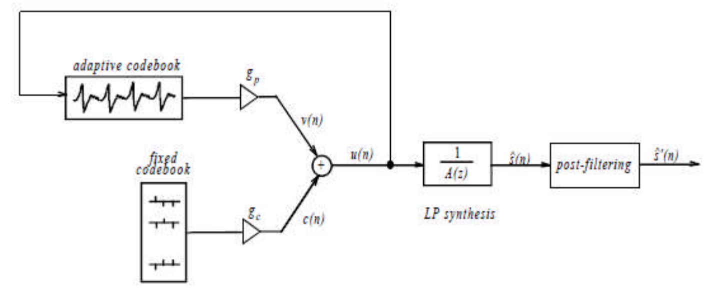
APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
[1b] an adaptive codebook;	The speech system using an analysis by synthesis approach on a speech signal in [Defendants' Accused Products] includes an adaptive codebook. The speech encoding system in [Defendants' Accused Products] includes an adaptive codebook. *See e.g.*, TS 26.090 at § 3.1 ("adaptive codebook: contains excitation vectors that are adapted for every subframe. The adaptive codebook is derived from the long-term filter state. The lag value can be viewed as an index into the adaptive codebook"), § 5.6.1 ("Adaptive codebook search is performed on a subframe basis. It consists of performing closed-loop pitch search, and then computing the adaptive codevector by interpolating the past excitation at the selected fractional pitch lag. The adaptive codebook parameters (or pitch parameters) are the delay and gain of the pitch filter. In the adaptive codebook approach for implementing the pitch filer, the excitation is repeated for delays less than the subframe length. In the search stage, the excitation is extended by the LP residual to simplify the closed-loop search."), Figure 2 (shown below).	To the extent this limitation is satisfied by the functionality accused in Plaintiff's 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard includes the following passages, which are virtually identical to the passages identified by WIAV: The CELP speech synthesis model is shown in figure 2. In this model, the excitation signal at the input of the short-term LP synthesis filter is constructed by adding two excitation vectors from adaptive and fixed (innovative) codebooks. The speech is synthesized by feeding the two properly chosen vectors from these codebooks through the short-term synthesis filter. The optimum excitation sequence in a codebook is chosen using an analysis-by-synthesis search procedure in which the error between the original and synthesized speech is minimized according to a perceptually weighted distortion measure. **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 4.3 (Mar. 1997)).** 5.6 Adaptive codebook search Adaptive codebook search is performed on a subframe basis. It consists of performing closed-loop pitch search, and then computing the adaptive codevector by interpolating the past excitation at the selected fractional pitch lag. The adaptive codebook parameters (or pitch parameters) are the delay and gain of the pitch filter. In the adaptive codebook approach for implementing the pitch filter, the excitation is repeated for delays less than the subframe length. In the search stage, the excitation is extended by the LP residual to simplify the closed-loop search. **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.6 (Mar. 1997)).**

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	 Figure 2: Simplified block diagram of the CELP synthesis model	 Figure 2: Simplified block diagram of the CELP synthesis model **GSM EFR 06.51, § 2 (DEC. 1997) (incorporating by reference GSM EFR 06.60, FIG. 2 (Mar. 1997)).**
[1c] a fixed codebook;	The speech system using an analysis by synthesis approach on a speech signal in [Defendants' Accused Products] includes a fixed codebook. The speech encoding system in [Defendants' Accused Products] includes an adaptive codebook. *See e.g.*, TS 26.090 at § 3.1 ("**fixed codebook:** The fixed codebook contains excitation vectors for speech synthesis filters. The contents of the codebook are non-adaptive (i.e., fixed). In the adaptive multi-rate codec, the fixed codebook is implemented using an algebraic codebook."), § 5.7.2 ("The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the	To the extent this limitation is satisfied by the functionality accused in Plaintiff's 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard includes the following passages, which are virtually identical to the passages identified by WIAV: The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive codebook contribution. That is: $x_2(n)=x(n)-\hat{g}_p y(n), \quad n=0,\ldots,39,$ (34) **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.6 (Mar. 1997)).**

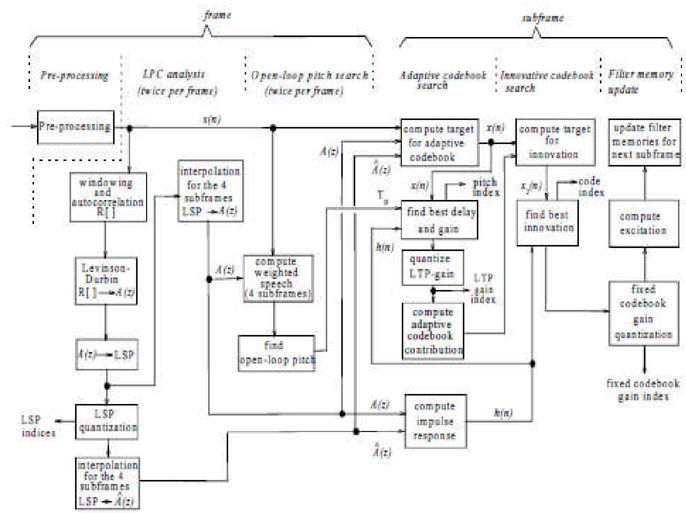
APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	adaptive codebook contribution."), Figure 2 (shown above).	 Figure 2: Simplified block diagram of the CELP synthesis model **GSM EFR 06.51, § 2 (DEC. 1997) (incorporating by reference GSM EFR 06.60, FIG. 2 (Mar. 1997)).**
[1d] a processing circuit that sequentially identifies a first gain applied to the adaptive codebook and a second gain applied to the fixed codebook; and	The speech system using an analysis by synthesis approach on a speech signal in [Defendants' Accused Products] includes a processing circuit that sequentially identifies a first gain applied to the adaptive codebook and a second gain applied to the fixed codebook. The speech system in [Defendants' Accused Products] products includes a processing circuit (Speech Encoder/Decoder) that identifies a first gain g_p for the adaptive codebook and a second gain g_c for the fixed codebook. As shown in the formulas below, the gains are identified sequentially. The adaptive codebook gain g_p is calculated first, then the target signal $x_2(n)$ is calculated using the quantized $\hat{g}_p$, and then the fixed codebook gain g_c is calculated based on the target	To the extent this limitation is satisfied by the functionality accused in Plaintiff's 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard includes the following passages, which are virtually identical to the passages identified by WIAV:

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	signal $x_2(n)$. *See e.g.*, TS 26.090 at § 5.6.1 and § 5.7.2. The adaptive codebook gain is then found by: $g_p = \frac{\sum_{n=0}^{39} x(n)y(n)}{\sum_{n=0}^{39} y(n)y(n)}, \quad \text{bounded by} \quad 0 \le g_p \le 1.2 \qquad (41)$ where $y(n) = v(n)* h(n)$ is the filtered adaptive codebook vector (zero state response of $H(z)W(z)$ to $v(n)$). The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive codebook contribution. That is: $x_2(n) = x(n) - \hat{g}_p y(n), \quad n = 0,\ldots,39 \ (42)$ where $y(n) = v(n)*h(n)$ is the filtered adaptive codebook vector and is the quantified $\hat{g}_p$ adaptive codebook gain. … The fixed codebook gain is then found by:	The adaptive codebook gain is then found by: $g_p = \frac{\sum_{n=0}^{39} x(n)y(n)}{\sum_{n=0}^{39} y(n)y(n)}, \quad \text{bounded by} \quad 0 \le g_p \le 1.2, \qquad (33)$ where $y(n) = v(n) * h(n)$ is the filtered adaptive codebook vector (zero state response of $H(z)W(z)$ to $v(n)$). **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.6 (Mar. 1997)).** The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive codebook contribution. That is: $x_2(n) = x(n) - \hat{g}_p y(n), \quad n = 0,\ldots,39, \qquad (34)$ where $y(n) = v(n) * h(n)$ is the filtered adaptive codebook vector and $\hat{g}_p$ is the quantified adaptive codebook gain. If c_k is the algebraic codevector at index k, then the algebraic codebook is searched by maximizing the term: The fixed codebook gain is then found by: $g_c = \frac{\mathbf{x}_2^t \mathbf{z}}{\mathbf{z}^t \mathbf{z}} \qquad (43)$ where $\mathbf{x}_2$ is the target vector for fixed codebook search and $\mathbf{z}$ is the fixed codebook vector convolved with $h(n)$, $z(n) = \sum_{i=0}^{n} c(i)\, h(n-i), \quad n = 0,\ldots,39. \qquad (44)$ **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.7 (Mar. 1997)).**

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	$$g_c = \frac{\mathbf{x}_2^t \mathbf{z}}{\mathbf{z}^t \mathbf{z}} \qquad (51)$$ Where $\mathbf{x_2}$ is the target vector for fixed codebook search and **z** is the fixed codebook vector convolved with *h(n).* $$z(n) = \sum_{i=0}^{n} c(i)h(n-i), \qquad n = 0,\ldots,39. \qquad (52)$$ *See also e.g.*, TS 26.073, including g_pitch.c (excerpted below) and g_pitch.h.	 Figure 3: Simplified block diagram of the GSM enhanced full rate encoder **GSM EFR 06.51, § 2 (DEC. 1997) (incorporating by reference GSM EFR 06.60, FIG. 3 (Mar. 1997)).**

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	/*** * * FUNCTION: G_pitch * * PURPOSE: Compute the pitch (adaptive codebook) gain. * Result in Q14 (NOTE: 12.2 bit exact using Q12) * * DESCRIPTION: * The adaptive codebook gain is given by * * g = <x[], y[]> / <y[], y[]> * * where x[] is the target vector, y[] is the filtered adaptive * codevector, and <> denotes dot product. * The gain is limited to the range [0,1.2] (=0..19661 Q14) * ***/ Word16 G_pitch (/* o : Gain of pitch lag saturated to 1.2 */ enum Mode mode, /* i : AMR mode */ Word16 xn[], /* i : Pitch target. */ Word16 y1[], /* i : Filtered adaptive codebook. */ Word16 g_coeff[], /* i : Correlations need for gain quantization */ Word16 L_subfr /* i : Length of subframe. */) { Word16 i; Word16 xy, yy, exp_xy, exp_yy, gain; Word32 s; Word16 scaled_y1[L_SUBFR]; /* Usually dynamic allocation of (L_subfr) */	/*** ********************** * * FUNCTION: G_pitch * * PURPOSE: Compute the pitch (adaptive codebook) gain. Result in Q12 * * DESCRIPTION: * The adaptive codebook gain is given by * * g = <x[], y[]> / <y[], y[]> * * where x[] is the target vector, y[] is the filtered adaptive * codevector, and <> denotes dot product. * The gain is limited to the range [0,1.2] * *** **********************/ #include "typedef.h" #include "basic_op.h" #include "oper_32b.h" #include "count.h" #include "sig_proc.h" Word16 G_pitch (/* (o) : Gain of pitch lag saturated to 1.2 */ Word16 xn[], /* (i) : Pitch target. */ Word16 y1[], /* (i) : Filtered adaptive codebook. */

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	/* divide "yl[]" by 4 to avoid overflow */ for (i = 0; i < L_subfr; i++) { scaled_yl[i] = shr (yl[i], 2); move16 (); } /* Compute scalar product <yl[],yl[]> */ /* Q12 scaling / MR122 */ Overflow = 0; move16 (); s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, yl[i], yl[i]); } test (); if (Overflow == 0) /* Test for overflow */ { exp_yy = norm_l (s); yy = round (L_shl (s, exp_yy)); } else { s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, scaled_yl[i], scaled_yl[i]); } exp_yy = norm_l (s); yy = round (L_shl (s, exp_yy)); exp_yy = sub (exp_yy, 4); } /* Compute scalar product <xn[],yl[]> */ Overflow = 0; move16 (); s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++)	Word16 L_subfr /* : Length of subframe. */) { Word16 i; Word16 xy, yy, exp_xy, exp_yy, gain; Word32 s; Word16 scaled_y1[80]; /* Usually dynamic allocation of (L_subfr) */ /* divide by 2 "y1[]" to avoid overflow */ for (i = 0; i < L_subfr; i++) { scaled_y1[i] = shr (y1[i], 2); move16 (); } /* Compute scalar product <y1[],y1[]> */ s = 0L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, y1[i], y1[i]); } test (); if (L_sub (s, MAX_32) != 0L) /* Test for overflow */ { s = L_add (s, 1L); /* Avoid case of all zeros */ exp_yy = norm_l (s); yy = round (L_shl (s, exp_yy));

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	<pre>/* if(gain >1.2) gain = 1.2 */ test (); if (sub (gain, 19661) > 0) { gain = 19661; move16 (); } test (); if (sub(mode, MR122) == 0) { /* clear 2 LSBits */ gain = gain & 0xfffC; logic16 (); } return (gain); }</pre>	<pre>} else { s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, scaled_y1[i], scaled_y1[i]); } exp_yy = norm_l (s); yy = round (L_shl (s, exp_yy)); exp_yy = sub (exp_yy, 4); } /* Compute scalar product <xn[],y1[]> */ Overflow = 0; move16 (); s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { Carry = 0; move16 (); s = L_macNs (s, xn[i], y1[i]); test (); if (Overflow != 0) { break; } } test (); if (Overflow == 0)</pre>

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
		`` { exp_xy = norm_l (s); xy = round (L_shl (s, exp_xy)); } else { s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, xn[i], scaled_y1[i]); } exp_xy = norm_l (s); xy = round (L_shl (s, exp_xy)); exp_xy = sub (exp_xy, 2); } /* If (xy < 4) gain = 0 */ i = sub (xy, 4); test (); if (i < 0) return ((Word16) 0); /* compute gain = xy/yy */ xy = shr (xy, 1); /* Be sure xy < yy */ gain = div_s (xy, yy); i = add (exp_xy, 3 - 1); /* Denormalization of division */ ```

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
		i = sub (i, exp_yy); gain = shr (gain, i); /* if(gain >1.2) gain = 1.2 */ test (); if (sub (gain, 4915) > 0) { gain = 4915; move16 (); } return (gain); } **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference G_pitch.c in GSM EFR 06.53 (June 26, 1996)).**
[1e] the processing circuit identifies a gain reduction factor applied to the first gain identified, the gain reduction factor is used by the processing circuit to perform the	The processing circuit in [Defendants' Accused Products] identifies a gain reduction factor applied to the first gain identified, the gain reduction factor is used by the processing circuit to perform the identification of the second gain. The processing circuit (Speech Encoder/Decoder) in [Defendants' Accused Products] identifies a gain reduction factor (GP_{th}) that is applied to the adaptive codebook gain g_p, and used by the Speech Encoder/Decoder to identify the fixed codebook gain g_c. See e.g., TS 26.090 at § 5.6. 1.	To the extent this limitation is satisfied by the functionality accused in Plaintiff's 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard discloses the use of a threshold (1.2) to limit the gain, as shown by the following passages:

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
identification of the second gain.	The adaptive codebook gain is then found by: $$g_p = \frac{\sum_{n=0}^{39} x(n)y(n)}{\sum_{n=0}^{39} y(n)y(n)}, \quad \text{bounded by} \quad 0 \le g_p \le 1.2 \qquad (41)$$ where $y(n) = v(n) * h(n)$ is the filtered adaptive codebook vector (zero state response of $H(z)W(z)$ to $v(n)$). *See also e.g.*, at § 5.6.2 ("Adaptive codebook gain control (all modes)"). The average adaptive codebook gain is calculated if the LSP_flag is set and the unquantized adaptive codebook gain exceeds the gain threshold $GP_{th} = 0.95$. The average gain is calculated from the present unquantized gain and the quantized gains of the seven previous subframes. That is, $GP_{ave} = mean\{g_p(n), \hat{g}_p(n-1), \hat{g}_p(n-2), \ldots, \hat{g}_p(n-7)\}$, where n is the current subframe. If the average adaptive codebook gain exceeds the GP_{th}, the unquantized gain is limited to the threshold value and the GpC_flag is set to indicate the limitation. $if(GP_{ave} > GP_{th})$ $g_p = GP_{th}$ $GpC\_flag = 1$ $else$ $GpC\_flag = 0$	The adaptive codebook gain is then found by: $$g_p = \frac{\sum_{n=0}^{39} x(n)y(n)}{\sum_{n=0}^{39} y(n)y(n)}, \quad \text{bounded by} \quad 0 \le g_p \le 1.2, \qquad (33)$$ where $y(n) = v(n) * h(n)$ is the filtered adaptive codebook vector (zero state response of $H(z)W(z)$ to $v(n)$). **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.6 (Mar. 1997)).** The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive codebook contribution. That is: $$x_2(n) = x(n) - \hat{g}_p y(n), \quad n = 0, \ldots, 39, \qquad (34)$$ where $y(n) = v(n) * h(n)$ is the filtered adaptive codebook vector and $\hat{g}_p$ is the quantified adaptive codebook gain. If $\mathbf{c}_k$ is the algebraic codevector at index k, then the algebraic codebook is searched by maximizing the term: The fixed codebook gain is then found by: $$g_c = \frac{\mathbf{x}_2^t \mathbf{z}}{\mathbf{z}^t \mathbf{z}} \qquad (43)$$ where $\mathbf{x}_2$ is the target vector for fixed codebook search and $\mathbf{z}$ is the fixed codebook vector convolved with $h(n)$, $$z(n) = \sum_{i=0}^{n} c(i)h(n-i), \quad n = 0, \ldots, 39. \qquad (44)$$ **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.7 (Mar. 1997)).** `/* if(gain >1.2) gain = 1.2 */` `test ();` `if (sub (gain, 4915) > 0)` `{` `gain = 4915;          move16 ();` `}` `return (gain);`

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	As shown in the ANSI-C code, the value of the gain reduction factor $_{GPth}$ is either 0.95 or 0.85 depending on the bit rate mode. *See e.g.*, TS 26.073, including cl_ltp.c (excerpted below) and cnst.h. `/*************************************************************************` `*` `*  Function:   cl_ltp` `*  Purpose:    closed-loop fractional pitch search` `*` `**************************************************************************` `*/` `int cl_ltp (` `    clLtpState *clSt,    /* i/o : State struct    */` `    tonStabState *tonSt, /* i/o : State struct    */` `    enum Mode mode,      /* i   : coder mode      */`	} **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference G_pitch.c in GSM EFR 06.53 (June 26, 1996)).**

APPENDIX 7-A

U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	<pre> Word16 frameOffset, /* i : Offset to subframe */ Word16 T_op[], /* i : Open loop pitch lags */ Word16 *h1, /* i : Impulse response vector Q12 */ Word16 *exc, /* i/o : Excitation vector Q0 */ Word16 res2[], /* i/o : Long term prediction residual Q0 */ Word16 xn[], /* i : Target vector for pitch search Q0 */ Word16 lsp_flag, /* i : LSP resonance flag */ Word16 xn2[], /* o : Target vector for codebook search Q0 */ Word16 y1[], /* o : Filtered adaptive excitation Q0 */ Word16 *T0, /* o : Pitch delay (integer part) */ Word16 *T0_frac, /* o : Pitch delay (fractional part) */ Word16 *gain_pit, /* o : Pitch gain Q14 */ Word16 g_coeff[], /* o : Correlations between xn, y1, & y2 */ Word16 **anap, /* o : Analysis parameters */ Word16 *gp_limit /* o : pitch gain limit */) { Word16 i; Word16 index; Word32 L_temp; /* temporarily variable */ Word16 resu3; /* flag for upsample resolution */ Word16 gpc_flag; /*---* * Closed-loop fractional pitch search * *---*/ *T0 = Pitch_fr(clSt->pitchSt, mode, T_op, exc, xn, h1, L_SUBFR, frameOffset, T0_frac, &resu3, &index); move16 (); *(*anap)++ = index; move16 (); /*---* * - find unity gain pitch excitation (adapitve codebook entry) * * with fractional interpolation. * * - find filtered pitch exc. y1[]=exc[] convolve with h1[]) * * - compute pitch gain and limit between 0 and 1.2 * * - update target vector for codebook search * * - find LTP residual. * *---*/</pre>	

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	Pred_lt_3or6(exc, *T0, *T0_frac, L_SUBFR, resu3); Convolve(exc, h1, y1, L_SUBFR); /* gain_pit is Q14 for all modes */ *gain_pit = G_pitch(mode, xn, y1, g_coeff, L_SUBFR); move16 (); /* check if the pitch gain should be limit due to resonance in LPC filter */ gpc_flag = 0; move16 (); *gp_limit = MAX_16; move16 (); test (); test (); if ((lsp_flag != 0) && (sub(*gain_pit, GP_CLIP) > 0)) { gpc_flag = check_gp_clipping(tonSt, *gain_pit); move16 (); } /* special for the MR475, MR515 mode; limit the gain to 0.85 to */ /* cope with bit errors in the decoder in a better way. */ test (); test (); if ((sub (mode, MR475) == 0) || (sub (mode, MR515) == 0)) { test (); if (sub (*gain_pit, 13926) > 0) { *gain_pit = 13926; /* 0.85 in Q14 */ move16 (); } test (); if (gpc_flag != 0) { *gp_limit = GP_CLIP; move16 (); } } else { test (); if (gpc_flag != 0) { *gp_limit = GP_CLIP; move16 (); *gain_pit = GP_CLIP; move16 (); }	

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	`/* For MR122, gain_pit is quantized here and not in gainQuant */` `if (test (), sub(mode, MR122)==0)` `{` `*(*anap)++ = q_gain_pitch(MR122, *gp_limit, gain_pit,` `NULL, NULL);` `move16 ();` `}` `}` `/* update target vector und evaluate LTP residual */` `for (i = 0; i < L_SUBFR; i++) {` `L_temp = L_mult(y1[i], *gain_pit);` `L_temp = L_shl(L_temp, 1);` `xn2[i] = sub(xn[i], extract_h(L_temp)); move16 ();` `L_temp = L_mult(exc[i], *gain_pit);` `L_temp = L_shl(L_temp, 1);` `res2[i] = sub(res2[i], extract_h(L_temp)); move16 ();` `}` `return 0;` `}` Further, the Speech Encoder/Decoder uses the gain reduction factor to identify the fixed codebook gain gc. *See e.g.*, TS 26.090 at § 5.6.1 (equation 41), § 5.7.2 (equations 42, 51 and 52) and § 5.8; *see also e.g.*, cl_ltp.c (excerpted above) and cnst.h.	
2. The speech system of claim 1 wherein the gain reduction factor comprises an adaptive gain factor.	In the speech system using an analysis by synthesis approach on a speech signal in [Defendants' Accused Products], the gain reduction factor comprises an adaptive gain factor. In the speech system in [Defendants' Accused Products], the gain reduction factor *(GP_{th})* comprises an adaptive gain factor as the value of the gain reduction factor (*i.e.*, 0.95 or 0.85) adapts based on the	

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	encoding bit rate. *See e.g.*, TS 26.073, including cl_ltp.c (excerpted above) and cnst.h; *see also e.g.*, at TS 26.090 at § 5.6.2 ("Adaptive codebook gain control (all modes)").	
3. The speech system of claim 2 wherein the processing circuit identifies the adaptive gain factor by considering, at least in part, an encoding bit rate.	In the speech system using an analysis by synthesis approach on a speech signal in [Defendants' Accused Products], the processing circuit identifies the adaptive gain factor by considering at least in part, an encoding bit rate. In the speech system in [Defendants' Accused Products], the Speech Encoder/Decoder identifies the gain reduction factor **(**GP_{th}) (*i.e.*, 0.95 or 0.85) by considering the encoding bit rate. *See e.g.*, TS 26.073, including cl_ltp.c (excerpted above) and cnst.h.	
7[a]. A speech system using an analysis by synthesis approach on a speech signal, the speech system comprising:	According to publicly available documentation, [Defendants' Accused Products] implement a speech system using an analysis by synthesis approach on a speech signal. The 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Mandatory speech CODEC speech processing functions; AMR speech CODEC; General description ("TS 26.071") describes the mandatory speech codec speech processing functions for the Adaptive Multi-Rate ("AMR") speech codec. "The multi-rate speech coder is a single integrated speech codec with eight source rates from 4.75 kbit/s to 12.2 kbit/s, and a low	To the extent this limitation is satisfied by the functionality accused in Plaintiff's 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard includes the following passages, which are virtually identical to the passages identified by WIAV:

APPENDIX 7-A

U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	rate background noise encoding mode." TS 26.071 at § 4. The 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Mandatory Speech Codec speech processing functions; Adaptive Multi-Rate (AMR) speech codec; Transcoding functions ("TS 26.090") describes the mandatory transcoding functions for the Adaptive Multi-Rate ("AMR") speech codec. "Clauses 4.3 and 4.4 present a simplified description of the principles of the AMR codec encoding and decoding process respectively. In clause 4.5, the sequence and subjective importance of encoded parameters are given. Clause 5 presents the functional description of the AMR codec encoding, whereas clause 6 describes the decoding procedures. In clause 7, the detailed bit allocation of the AMR codec is tabulated." TS 26.090 at § 4; § 6 ("The function of the decoder consists of decoding the transmitted parameters (LP parameters, adaptive codebook vector, adaptive codebook gain, fixed codebook vector, fixed codebook gain) and performing synthesis to obtain the reconstructed speech. The reconstructed speech is then post-filtered and upscaled. The signal flow at the decoder is shown in figure 4."). The 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; ANSI-C code for the Adaptive Multi Rate (AMR) speech codec ("TS 26.073") "contains an electronic copy of the ANSI-C code for the Adaptive Multi-Rate	4.3 Principles of the GSM enhanced full rate speech encoder The codec is based on the code-excited linear predictive (CELP) coding model. A 10th order linear prediction (LP), or short-term, synthesis filter is used which is given by: **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 4.3 (Mar. 1997)).** The function of the decoder consists of decoding the transmitted parameters (LP parameters, adaptive codebook vector, adaptive codebook gain, fixed codebook vector, fixed codebook gain) and performing synthesis to obtain the reconstructed speech. The reconstructed speech is then post-filtered and upscaled. The signal flow at the decoder is shown in figure 4. **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference Section 6 GSM EFR 06.60 (Mar. 1997)).**

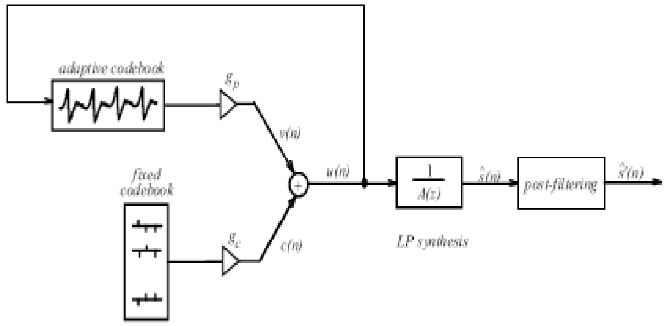
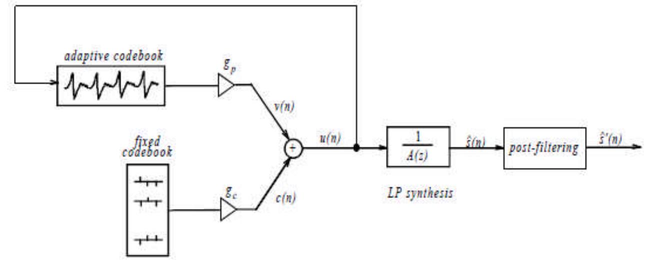
APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	codec. The ANSI-C code is necessary for a bit exact implementation of the Adaptive Multi Rate speech transcoder (TS 26.090 [2]), Voice Activity Detection (TS 26.094 [6]), comfort noise (TS 26.092 [4]), source controlled rate operation (TS 26.093 [5]) and example solutions for substituting and muting of lost frames (TS 26.091 [3]).” TS 26.071 at § 1. “In the case of discrepancy between the requirements described in the present document and the fixed point computational description (ANSI-C code) of these requirements contained in [TS 26.073], the description in [TS 26.073] will prevail.” TS 26.090 at § 1. [Defendants' Accused Products] include an Adaptive Multi-Rate (“AMR”) speech system that uses an analysis by synthesis approach for coding and decoding speech signal. *See e.g.*, TS 26.090 at § 4.3 (“The CELP speech synthesis model is shown in figure 2. In this model, the excitation signal at the input of the short-term LP synthesis filter is constructed by adding two excitation vectors from adaptive and fixed (innovative) codebooks. The speech is synthesized by feeding the two properly chosen vectors from these codebooks through the short-term synthesis filter. The optimum excitation sequence in a codebook is chosen using an analysis-by-synthesis search procedure in which the error between the original and synthesized speech is minimized according to a perceptually weighted distortion measure.”); Figure 2.	The CELP speech synthesis model is shown in figure 2. In this model, the excitation signal at the input of the short-term LP synthesis filter is constructed by adding two excitation vectors from adaptive and fixed (innovative) codebooks. The speech is synthesized by feeding the two properly chosen vectors from these codebooks through the short-term synthesis filter. The optimum excitation sequence in a codebook is chosen using an analysis-by-synthesis search procedure in which the error between the original and synthesized speech is minimized according to a perceptually weighted distortion measure. **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 4.3 (Mar. 1997)).**

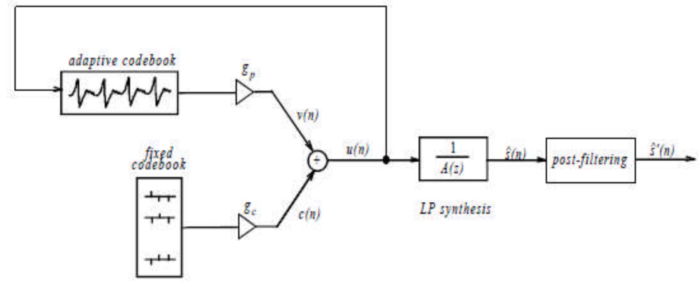
APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
[7b] a adaptive codebook;	The speech system using an analysis by synthesis approach on a speech signal in [Defendants' Accused Products] includes an adaptive codebook. The speech system in [Defendants' Accused Products] includes an adaptive codebook. *See e.g.,* TS 26.090 at § 3.1 **("adaptive codebook:** contains excitation vectors that are adapted for every subframe. The adaptive codebook is derived from the long-term filter state. The lag value can be viewed as an index into the adaptive codebook"); § 5.6.1 ("Adaptive codebook search is performed on a subframe basis. It consists of performing closed-loop pitch search, and then computing the adaptive codevector by interpolating the past excitation at the selected fractional pitch lag. The adaptive codebook parameters (or pitch parameters) are the delay and gain of the pitch filter. In the adaptive codebook approach for implementing the pitch filter, the excitation is repeated for delays less than the subframe length. In the search stage, the excitation is extended by the LP residual to simplify the closed-loop search."); Figure 2 (shown below).	To the extent this limitation is satisfied by the functionality accused in Plaintiff's 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard includes the following passages, which are virtually identical to the passages identified by WIAV: The CELP speech synthesis model is shown in figure 2. In this model, the excitation signal at the input of the short-term LP synthesis filter is constructed by adding two excitation vectors from adaptive and fixed (innovative) codebooks. The speech is synthesized by feeding the two properly chosen vectors from these codebooks through the short-term synthesis filter. The optimum excitation sequence in a codebook is chosen using an analysis-by-synthesis search procedure in which the error between the original and synthesized speech is minimized according to a perceptually weighted distortion measure. **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 4.3 (Mar. 1997)).** 5.6 Adaptive codebook search Adaptive codebook search is performed on a subframe basis. It consists of performing closed-loop pitch search, and then computing the adaptive codevector by interpolating the past excitation at the selected fractional pitch lag. The adaptive codebook parameters (or pitch parameters) are the delay and gain of the pitch filter. In the adaptive codebook approach for implementing the pitch filter, the excitation is repeated for delays less than the subframe length. In the search stage, the excitation is extended by the LP residual to simplify the closed-loop search. **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.6 (Mar. 1997)).**

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV’s Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	 Figure 2: Simplified block diagram of the CELP synthesis model	 Figure 2: Simplified block diagram of the CELP synthesis model **GSM EFR 06.51 (DEC. 1997) (incorporating by reference, GSM EFR 06.60, FIG. 2 (Mar. 1997)).**
[7c] a fixed codebook;	The speech system using an analysis by synthesis approach on a speech signal in [Defendants’ Accused Products] includes a fixed codebook. The speech system in [Defendants’ Accused Products] includes an adaptive codebook. See e.g., TS 26.090 at § 3.1 (“**fixed codebook:** The fixed codebook contains excitation vectors for speech synthesis filters. The contents of the codebook are non-adaptive (i.e., fixed). In the adaptive multi-rate codec, the fixed codebook is implemented using an algebraic codebook.”); § 5.7.2 (“The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive codebook	To the extent this limitation is satisfied by the functionality accused in Plaintiff’s 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard includes the following passages, which are virtually identical to the passages identified by WIAV: The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive codebook contribution. That is: $x_2(n)=x(n)-\hat{g}_p y(n), \quad n=0,\ldots,39,$ (34) **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.6 (Mar. 1997)).**

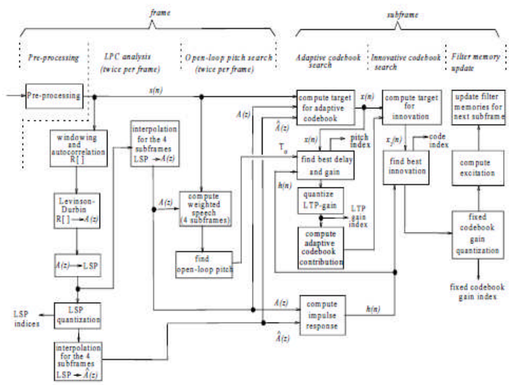
APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	contribution."); Figure 2 (shown above).	 Figure 2: Simplified block diagram of the CELP synthesis model **GSM EFR 06.51 (DEC. 1997) (incorporating by reference, GSM EFR 06.60, FIG. 2 (Mar. 1997)).**
[7d] a processing circuit that generates a first contribution from the adaptive codebook and a second contribution from the fixed codebook; and	The speech system using an analysis by synthesis approach on a speech signal in [Defendants' Accused Products] includes a processing circuit that generates a first contribution from the adaptive codebook and a second contribution from the fixed codebook. The speech system in [Defendants' Accused Products] includes a processing circuit (Speech Encoder/Decoder) that generates a first contribution from the adaptive codebook (i.e., adaptive codebook vector *v(n)* and gain g_p) and a second contribution from the fixed codebook (i.e., fixed codebook vector *c(n)* and gain g_c). *See e.g.*, TS 26.090 at § 5.6.1. Once the fractional pitch lag is determined, the adaptive codebook vector *v(n)* is computed by	To the extent this limitation is satisfied by the functionality accused in Plaintiff's 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard includes the following passages, which are virtually identical to the passages identified by WIAV:

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	interpolating the past excitation signal *u(n)* at the given integer delay *k* and phase (fraction) *t* : $v(n)=\sum_{i=0}^{9}u(n-k-i)b_{60}(t+i\cdot 6)+\sum_{i=0}^{9}u(n-k+1+i)b_{60}(6-t+i\cdot 6),\quad n=0,\ldots,39,\ t=0,\ldots,5.$ (40) … The adaptive codebook gain is then found by: $g_p=\frac{\sum_{n=0}^{39}x(n)y(n)}{\sum_{n=0}^{39}y(n)y(n)},\quad \text{bounded by}\quad 0\le g_p\le 1.2\qquad (41)$ where $y(n) = v(n) * h(n)$ is the filtered adaptive codebook vector (zero state response of $H(z)W(z)$ to $v(n)$). *See also e.g.*, TS 26.090 at § 5.7.2. The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive codebook contribution. That is:	Once the fractional pitch lag is determined, the adaptive codebook vector $v(n)$ is computed by interpolating the past excitation signal $u(n)$ at the given integer delay k and phase (fraction) t : $v(n)=\sum_{i=0}^{9}u(n-k-i)b_{60}(t+i\cdot 6)+\sum_{i=0}^{9}u(n-k+1+i)b_{60}(6-t+i\cdot 6),\quad n=0,\ldots,39,\ t=0,\ldots,5.\quad (32)$ . . . The adaptive codebook gain is then found by: $g_p=\frac{\sum_{n=0}^{39}x(n)y(n)}{\sum_{n=0}^{39}y(n)y(n)},\quad \text{bounded by}\quad 0\le g_p\le 1.2,\qquad (33)$ where $y(n)=v(n)*h(n)$ is the filtered adaptive codebook vector (zero state response of $H(z)W(z)$ to $v(n)$). **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.6 (Mar. 1997)).** The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive codebook contribution. That is: $x_2(n)=x(n)-\hat{g}_p y(n),\quad n=0,\ldots,39,\qquad (34)$ where $y(n)=v(n)*h(n)$ is the filtered adaptive codebook vector and $\hat{g}_p$ is the quantified adaptive codebook gain. If c_k is the algebraic codevector at index k, then the algebraic codebook is searched by maximizing the term:

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	$$x_2(n) = x(n) - \hat{g}_p y(n), \quad n = 0,\ldots,39 \ (42)$$ where $y(n) = v(n)*h(n)$ is the filtered adaptive codebook vector and $\hat{g}_p$ is the quantified adaptive codebook gain. ... The fixed codebook gain is then found by: $$g_c = \frac{\mathbf{x}_2^t \mathbf{z}}{\mathbf{z}^t \mathbf{z}} \qquad (51)$$ Where $\mathbf{x}_2$ is the target vector for fixed codebook search and $\mathbf{z}$ is the fixed codebook vector convolved with $h(n)$, $$z(n) = \sum_{i=0}^{n} c(i)h(n-i), \qquad n = 0,\ldots,39. \quad (52)$$ *See also e.g.,* TS 26.073, including g_pitch.c (excerpted below) and g_pitch.h. `/*************************************************************************` `*` `* FUNCTION: G_pitch` `*` `* PURPOSE: Compute the pitch (adaptive codebook) gain.` `*          Result in Q14 (NOTE: 12.2 bit exact using Q12)` `*` `* DESCRIPTION:`	The fixed codebook gain is then found by: $$g_c = \frac{\mathbf{x}_2^t \mathbf{z}}{\mathbf{z}^t \mathbf{z}} \qquad (43)$$ where $\mathbf{x}_2$ is the target vector for fixed codebook search and $\mathbf{z}$ is the fixed codebook vector convolved with $h(n)$, $$z(n) = \sum_{i=0}^{n} c(i)h(n-i), \qquad n = 0,\ldots,39. \qquad (44)$$ **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.7 (Mar. 1997)).**  Figure 3: Simplified block diagram of the GSM enhanced full rate encoder **GSM EFR 06.51 (DEC. 1997) (incorporating by reference, GSM EFR 06.60, FIG. 3 (Mar. 1997)).**

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	* The adaptive codebook gain is given by * * g = <x[], y[]> / <y[], y[]> * * where x[] is the target vector, y[] is the filtered adaptive * codevector, and <> denotes dot product. * The gain is limited to the range [0,1.2] (=0..19661 Q14) * **/ Word16 G_pitch (/* o : Gain of pitch lag saturated to 1.2 */ enum Mode mode, /* i : AMR mode */ Word16 xn[], /* i : Pitch target. */ Word16 y1[], /* i : Filtered adaptive codebook. */ Word16 g_coeff[], /* i : Correlations need for gain quantization */ Word16 L_subfr /* i : Length of subframe. */) { Word16 i; Word16 xy, yy, exp_xy, exp_yy, gain; Word32 s; Word16 scaled_y1[L_SUBFR]; /* Usually dynamic allocation of (L_subfr) */ /* divide "y1[]" by 4 to avoid overflow */ for (i = 0; i < L_subfr; i++) { scaled_y1[i] = shr (y1[i], 2); move16 (); } /* Compute scalar product <y1[],y1[]> */ /* Q12 scaling / MR122 */ Overflow = 0; move16 (); s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, y1[i], y1[i]); } test (); if (Overflow == 0) /* Test for overflow */	/*** ********************* * * FUNCTION: G_pitch * * PURPOSE: Compute the pitch (adaptive codebook) gain. Result in Q12 * * DESCRIPTION: * The adaptive codebook gain is given by * * g = <x[], y[]> / <y[], y[]> * * where x[] is the target vector, y[] is the filtered adaptive * codevector, and <> denotes dot product. * The gain is limited to the range [0,1.2] * *** *********************/ #include "typedef.h" #include "basic_op.h" #include "oper_32b.h" #include "count.h" #include "sig_proc.h" Word16 G_pitch (/* (o) : Gain of pitch lag saturated to 1.2 */ Word16 xn[], /* (i) : Pitch target. */ Word16 y1[], /* (i) : Filtered adaptive codebook. */

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
		Word16 L_subfr /* : Length of subframe. */) { Word16 i; Word16 xy, yy, exp_xy, exp_yy, gain; Word32 s; Word16 scaled_y1[80]; /* Usually dynamic allocation of (L_subfr) */ /* divide by 2 "y1[]" to avoid overflow */ for (i = 0; i < L_subfr; i++) { scaled_y1[i] = shr (y1[i], 2); move16 (); } /* Compute scalar product <y1[],y1[]> */ s = 0L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, y1[i], y1[i]); } test (); if (L_sub (s, MAX_32) != 0L) /* Test for overflow */ { s = L_add (s, 1L); /* Avoid case of all zeros */ exp_yy = norm_l (s); yy = round (L_shl (s, exp_yy));

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	{ exp_yy = norm_l (s); yy = round (L_shl (s, exp_yy)); } else { s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, scaled_y1[i], scaled_y1[i]); } exp_yy = norm_l (s); yy = round (L_shl (s, exp_yy)); exp_yy = sub (exp_yy, 4); } /* Compute scalar product <xn[],y1[]> */ Overflow = 0; move16 (); s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac(s, xn[i], y1[i]); } test (); if (Overflow == 0) { exp_xy = norm_l (s); xy = round (L_shl (s, exp_xy)); } else { s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, xn[i], scaled_y1[i]); } exp_xy = norm_l (s); xy = round (L_shl (s, exp_xy)); exp_xy = sub (exp_xy, 2);	} else { s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, scaled_y1[i], scaled_y1[i]); } exp_yy = norm_l (s); yy = round (L_shl (s, exp_yy)); exp_yy = sub (exp_yy, 4); } /* Compute scalar product <xn[],y1[]> */ Overflow = 0; move16 (); s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { Carry = 0; move16 (); s = L_macNs (s, xn[i], y1[i]); test (); if (Overflow != 0) { break; } } test (); if (Overflow == 0)

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	} g_coeff[0] = yy; move16 (); g_coeff[1] = sub (15, exp_yy); move16 (); g_coeff[2] = xy; move16 (); g_coeff[3] = sub (15, exp_xy); move16 (); /* If (xy < 4) gain = 0 */ i = sub (xy, 4); test (); if (i < 0) return ((Word16) 0); /* compute gain = xy/yy */ xy = shr (xy, 1); /* Be sure xy < yy */ gain = div_s (xy, yy); i = sub (exp_xy, exp_yy); /* Denormalization of division */ gain = shr (gain, i); /* if(gain >1.2) gain = 1.2 */ test (); if (sub (gain, 19661) > 0) { gain = 19661; move16 (); } test (); if (sub(mode, MR122) == 0) { /* clear 2 LSBits */ gain = gain & 0xfffC; logic16 (); } return (gain); }	{ exp_xy = norm_l (s); xy = round (L_shl (s, exp_xy)); } else { s = 1L; move32 (); /* Avoid case of all zeros */ for (i = 0; i < L_subfr; i++) { s = L_mac (s, xn[i], scaled_y1[i]); } exp_xy = norm_l (s); xy = round (L_shl (s, exp_xy)); exp_xy = sub (exp_xy, 2); } /* If (xy < 4) gain = 0 */ i = sub (xy, 4); test (); if (i < 0) return ((Word16) 0); /* compute gain = xy/yy */ xy = shr (xy, 1); /* Be sure xy < yy */ gain = div_s (xy, yy); i = add (exp_xy, 3 - 1); /* Denormalization of division */

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
		i = sub (i, exp_yy); gain = shr (gain, i); /* if(gain >1.2) gain = 1.2 */ test (); if (sub (gain, 4915) > 0) { gain = 4915; move16 (); } return (gain); } **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference G_pitch.c in GSM EFR 06.53 (June 26, 1996)).**
[7e] the processing circuit applying gain reduction to the first contribution from the adaptive codebook then regenerating the second contribution from the fixed codebook.	The processing circuit in [Defendants' Accused Products] applies gain reduction to the first contribution from the adaptive codebook then regenerates the second contribution from the fixed codebook. The processing circuit (Speech Encoder/Decoder) in [Defendants' Accused Products] applies gain reduction to the first contribution from the adaptive codebook (*i.e.*, adaptive codebook vector *v(n)* and gain g_p) by using GP_{th} (either 0.95 or 0.85). *See e.g.*, TS 26.090 at § 5.6.1.	To the extent this limitation is satisfied by the functionality accused in Plaintiff's 11/23/09 infringement contentions, this limitation is disclosed in the prior art. For example, without limitation, a prior art version of the GSM EFR codec standard discloses the use of a threshold (1.2) to limit the gain, as shown by the following passages:

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	The adaptive codebook gain is then found by: $$g_p = \frac{\sum_{n=0}^{39} x(n)y(n)}{\sum_{n=0}^{39} y(n)y(n)}, \quad \text{bounded by} \quad 0 \le g_p \le 1.2 \quad ($$ where $y(n) = v(n)* h(n)$ is the filtered adaptive codebook vector (zero state response of $H(z)W(z)$ to $v(n)$). *See also e.g.*, at § 5.6.2 ("Adaptive codebook gain control (all modes)"). The average adaptive codebook gain is calculated if the *LSP_flag* is set and the unquantized adaptive codebook gain exceeds the gain threshold $GP_{th} = 0.95$ [redacted]. The average gain is calculated from the present unquantized gain and the quantized gains of the seven previous subframes. That is, $GP_{ave} = mean\{g_p(n), \hat{g}_p(n-1), \hat{g}_p(n-2), \ldots, \hat{g}_p(n-7)\}$ [redacted], where n is the current subframe. If the average adaptive codebook gain exceeds the GP_{th} [redacted], the unquantized gain is limited to the threshold value and the *GpC_flag* is set to indicate the limitation.	The adaptive codebook gain is then found by: $$g_p = \frac{\sum_{n=0}^{39} x(n)y(n)}{\sum_{n=0}^{39} y(n)y(n)}, \quad \text{bounded by} \quad 0 \le g_p \le 1.2, \qquad (33)$$ where $y(n)=v(n)*h(n)$ is the filtered adaptive codebook vector (zero state response of $H(z)W(z)$ to $v(n)$). **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.6 (Mar. 1997)).** The algebraic codebook is searched by minimizing the mean square error between the weighted input speech and the weighted synthesized speech. The target signal used in the closed-loop pitch search is updated by subtracting the adaptive codebook contribution. That is: $$x_2(n)=x(n)-\hat{g}_p y(n), \quad n=0,\ldots,39, \qquad (34)$$ where $y(n)=v(n)*h(n)$ is the filtered adaptive codebook vector and $\hat{g}_p$ is the quantified adaptive codebook gain. If $\mathbf{c}_k$ is the algebraic codevector at index k, then the algebraic codebook is searched by maximizing the term: The fixed codebook gain is then found by: $$g_c = \frac{\mathbf{x}_2^t \mathbf{z}}{\mathbf{z}^t \mathbf{z}} \qquad (43)$$ where $\mathbf{x}_2$ is the target vector for fixed codebook search and $\mathbf{z}$ is the fixed codebook vector convolved with $h(n)$, $$z(n)=\sum_{i=0}^{n} c(i)h(n-i), \quad n=0,\ldots,39. \qquad (44)$$ **GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference GSM EFR 06.60, § 5.7 (Mar. 1997)).** `/* if(gain >1.2) gain = 1.2 */` `test ();` `if (sub (gain, 4915) > 0)` `{` `gain = 4915;      move16 ();` `}` `return (gain);` `}`

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	$if\,(GP_{ave} > GP_{th})$ $\quad g_p = GP_{th}$ $\quad GpC\_flag = 1$ $else$ $\quad GpC\_flag = 0$ As shown in the ANSI-C code, the value of the gain reduction factor GP_{th} is either 0.95 or 0.85 depending on the bit rate mode. *See e.g.,* TS 26.073, including cl_ltp.c (excerpted below) and cnst.h.	**GSM EFR 06.51, § 2 (Dec. 1997) (incorporating by reference G_pitch.c in GSM EFR 06.53 (June 26, 1996)).**

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	<pre>/*** * * Function: cl_ltp * Purpose: closed-loop fractional pitch search * ** */ int cl_ltp (clLtpState *clSt, /* i/o : State struct */ tonStabState *tonSt, /* i/o : State struct */ enum Mode mode, /* i : coder mode */ Word16 frameOffset, /* i : Offset to subframe */ Word16 T_op[], /* i : Open loop pitch lags */ Word16 *h1, /* i : Impulse response vector Q12 */ Word16 *exc, /* i/o : Excitation vector Q0 */ Word16 res2[], /* i/o : Long term prediction residual Q0 */ Word16 xn[], /* i : Target vector for pitch search Q0 */ Word16 lsp_flag, /* i : LSP resonance flag */ Word16 xn2[], /* o : Target vector for codebook search Q0 */ Word16 y1[], /* o : Filtered adaptive excitation Q0 */ Word16 *T0, /* o : Pitch delay (integer part) */ Word16 *T0_frac, /* o : Pitch delay (fractional part) */ Word16 *gain_pit, /* o : Pitch gain Q14 */ Word16 g_coeff[], /* o : Correlations between xn, y1, & y2 */ Word16 **anap, /* o : Analysis parameters */ Word16 *gp_limit /* o : pitch gain limit */)</pre>	

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	<pre>{ Word16 i; Word16 index; Word32 L_temp; /* temporarily variable */ Word16 resu3; /* flag for upsample resolution */ Word16 gpc_flag; /*--* * Closed-loop fractional pitch search * *--*/ *T0 = Pitch_fr(clSt->pitchSt, mode, T_op, exc, xn, h1, L_SUBFR, frameOffset, T0_frac, &resu3, &index); move16 (); *(*anap)++ = index; move16 (); /*---* * - find unity gain pitch excitation (adapitve codebook entry) * * with fractional interpolation. * * - find filtered pitch exc. y1[]=exc[] convolve with h1[]) * * - compute pitch gain and limit between 0 and 1.2 * * - update target vector for codebook search * * - find LTP residual. * *---*/ Pred_lt_3or6(exc, *T0, *T0_frac, L_SUBFR, resu3); Convolve(exc, h1, y1, L_SUBFR); /* gain_pit is Q14 for all modes */ *gain_pit = G_pitch(mode, xn, y1, g_coeff, L_SUBFR); move16 (); /* check if the pitch gain should be limit due to resonance in LPC filter */ gpc_flag = 0; move16 (); *gp_limit = MAX_16; move16 (); test (); test (); if ((lsp_flag != 0) && (sub(*gain_pit, GP_CLIP) > 0)) {</pre>	

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	<pre> gpc_flag = check_gp_clipping(tonSt, *gain_pit); move16 (); } /* special for the MR475, MR515 mode; limit the gain to 0.85 to */ /* cope with bit errors in the decoder in a better way. */ test (); test (); if ((sub (mode, MR475) == 0) || (sub (mode, MR515) == 0)) { test (); if (sub (*gain_pit, 13926) > 0) { *gain_pit = 13926; /* 0.85 in Q14 */ move16 (); } test (); if (gpc_flag != 0) { *gp_limit = GP_CLIP; move16 (); } } else { test (); if (gpc_flag != 0) { *gp_limit = GP_CLIP; move16 (); *gain_pit = GP_CLIP; move16 (); } /* For MR122, gain_pit is quantized here and not in gainQuant */ if (test (), sub(mode, MR122)==0) { *(*anap)++ = q_gain_pitch(MR122, *gp_limit, gain_pit, NULL, NULL); move16 (); } } /* update target vector und evaluate LTP residual */ for (i = 0; i < L_SUBFR; i++) { L_temp = L_mult(y1[i], *gain_pit); L_temp = L_shl(L_temp, 1); xn2[i] = sub(xn[i], extract_h(L_temp)); move16 (); L_temp = L_mult(exc[i], *gain_pit);</pre>	

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
	<pre> L_temp = L_shl(L_temp, 1); res2[i] = sub(res2[i], extract_h(L_temp)); move16 (); } return 0; }</pre> Further, after applying gain reduction, the Speech Encoder/Decoder regenerates the fixed codebook contribution from the fixed codebook (*i.e.*, fixed codebook vector *c(n)* and gain *gc*). *See e.g.*, TS 26.090 at §§ 5.6.1 (equation 41) and 5.7.2 (equations 42, 51 and 52); *see also e.g.*, TS 26.073, including cl_ltp.c (excerpted above) and cnst.h.	
8. The speech system of claim 7 wherein the gain reduction comprises application of a gain factor.	In the speech system using an analysis by synthesis approach on a speech signal in [Defendants' Accused Products], the gain reduction comprises application of a gain factor. In the speech system in [Defendants' Accused Products], the gain reduction comprises an application of a gain factor *(GP_{th})* (*i.e.*, 0.95 or 0.85). *See e.g.*, TS 26.073, including cl_ltp.c (excerpted above) and cnst.h; *see also e.g.*, TS 26.090 at § 5.6.2 ("Adaptive codebook gain control (all modes)").	
9. The speech system of claim 8 wherein the processing circuit	In the speech system using an analysis by synthesis approach on a speech signal in [Defendants' Accused Products], the processing circuit identifies the gain factor by considering at least in part, an encoding bit rate.	

APPENDIX 7-A
U.S. PATENT 6,104,992 IS INVALID UNDER 35 U.S.C. § 102 IN LIGHT OF GSM EFR 06.51 (Dec. 1997)

Claims	WiAV's Infringement Contentions	INVALIDITY BASED ON GSM EFR 06.51 (Dec. 1997)
identifies the gain factor by considering an encoding bit rate.	In the speech system in [Defendants' Accused Products], the Speech Encoder/Decoder identifies the gain reduction factor (GP_{th}) (*i.e.*, 0.95 or 0.85) by considering the encoding bit rate. *See e.g.*, TS 26.073, including cl_ltp.c (excerpted above) and cnst.h; *see also e.g.*, TS 26.090 at § 5.6.2 ("Adaptive codebook gain control (all modes)").	